Curso de Controladores Lógicos Programáveis



Faculdade de Engenharia

Laboratório de Engenharia Elétrica

Programa Prodenge / Sub-Programa Reenge
Universidade do Estado do Rio de Janeiro

AGRADECIMENTOS

Esta apostila é um produto do esforço do Laboratório de Engenharia Elétrica e da Faculdade de Engenharia da UERJ e mostra a tenacidade de alguns profissionais dedicados a causa da educação tecnológica e a crença de que é possível desenvolver um ambiente que estimule a criatividade e iniciativa dos alunos.

Muitas pessoas contribuíram para o seu desenvolvimento. No Laboratório, gostaríamos de agradecer aos bolsistas de iniciação científica, Joana Figueiredo Konte, Jorge Luís Pinheiro Teixeira, Pat Evie Alves; estagiários, Luciana Faletti, Hélio Justino Mattos Filho, Marcelo da Silveira Sobrinho, Robson Ramirez, César Cunha de Souza, Karla Karraz Walder , Flávia Delduque Lima ; funcionários André Vallim, Jair Medeiros Júnior, Marcos Augusto Mafra, Paulo Bulkool, José Emílio Gomes, Antônio Marcos Medeiros, Alberto Avelar Santiago, João Elias Souza, Luiz Roberto Fagundes, Sueli Ferreira, Carla Aparecida C. de Almeida. Gostaríamos de agradecer a direção da Faculdade de Engenharia e em especial ao diretor desta, Nival Nunes de Almeida, pelo incentivo dado a todas as atividades do LEE.

Esta apostila é parte do material didático dos cursos de Controladores Lógicos Programáveis - CLPs que serão ministrados no âmbito do Laboratório de Engenharia Elétrica. Agradecemos ao CNPq, que é o órgão financiador do programa PRODENGE – sub-programa REENGE, do qual faz parte este curso, pelo apoio financeiro recebido.

Este trabalho constitui uma ampliação das notas sobre programação do CLP S7-200 na versão DOS do Step7, elaborada pelos funcionários Jair Medeiros Júnior e Marcos Augusto Mafra, que foram utilizadas em treinamento interno.

Bernardo Severo da Silva Filho

Chefe do Lab. de Engenharia Elétrica

Orientador

SUMÁRIO

INTRODUÇÃO

Os Controladores Lógicos Programáveis ou CLPs, são equipamentos eletrônicos utilizados em sistemas de automação flexível. São ferramentas de trabalho muito úteis e versáteis para aplicações em sistemas de acionamentos e controle, e por isso são utilizados em grande escala no mercado industrial. Permitem desenvolver e alterar facilmente a lógica para acionamento das saídas em função das entradas. Desta forma, podemos associar diversos sinais de entrada para controlar diversos atuadores ligados nos pontos de saída.

1. Mercado Atual – Rápido e Flexível

A roda viva da atualização, da qual fazemos parte, movimenta e impulsiona o mercado mundial atualmente. Os profissionais buscam conhecimentos para se tornarem mais versáteis, adequando-se às necessidades das empresas, que por sua vez, buscam maior variedade e rapidez de produção para atender ao cliente, que se torna cada vez mais exigente.

As empresas estão se reorganizando para atender as necessidades atuais de aumento de produtividade, flexibilidade e redução de custos. Destas necessidades surgiram as necessidades de os equipamentos se adequarem rapidamente às alterações de configurações necessárias para produzirem diversos modelos de produtos, com pequenas alterações entre si.

2. Automação

Em princípio, qualquer grandeza física pode ser controlada, isto é, pode Ter seu valor intencionalmente alterado. Obviamente, há limitações práticas; uma das inevitáveis é a restrição da energia de que dispomos para afetar os fenômenos: por exemplo, a maioria das variáveis climatológicas poder ser medida mas não controlada, por causa da ordem de grandeza da energia envolvida.

O controle manual implica em se ter um operador presente ao processo criador de uma variável física e que, de acordo com alguma regra de seu conhecimento, opera um aparelho qualquer (válvula, alavanca, chave, ...), que por sua vez produz alterações naquela variável.

No início da industrialização, os processos industriais utilizavam o máximo da força da mão-de-obra. A produção era composta por etapas ou estágios, nos quais as pessoas desenvolviam sempre as mesmas funções, especializando-se em certa tarefa ou etapa da produção. Assim temos o princípio da produção seriada.

O mesmo ocorria com as máquinas de produção, que eram específicas para uma aplicação, o que impedia seu uso em outras etapas da produção, mesmo que tivesse características muito parecidas.

Com o passar do tempo e a valorização do trabalhador, foi preciso fazer algumas alterações nas máquinas e equipamentos, de forma a resguardar a mão-de-obra de algumas funções inadequadas à estrutura física do homem. A máquina passou a fazer o trabalho mais pesado e o homem, a supervisioná-la.

Com a finalidade de garantir o controle do sistema de produção, foram colocados **sensores** nas máquinas para monitorar e indicar as condições do processo. O controle só é garantido com o acionamento de **atuadores** a partir do processamento das informações coletadas pelos sensores.

O controle diz-se *automático* quando uma parte, ou a totalidade, das funções do operador é realizada por um equipamento, freqüente mas não necessariamente eletrônico.

Controle *automático por realimentação* é o equipamento automático que age sobre o elemento de controle, baseando-se em informações de medida da variável controlada. Como exemplo: o controle de temperatura de um refrigerador.

O controle *automático por programa* envolve a existência de um programa de ações, que se cumpre com base no decurso do tempo ou a partir de modificações eventuais em variáveis externas ao sistema. No primeiro caso temos um programa temporal e no segundo um programa lógico.

Automatizar um sistema, tornou-se muito mais viável à medida que a Eletrônica avançou e passou a dispor de circuitos capazes de realizar funções lógicas e aritméticas com os sinais de entrada e gerar respectivos sinais de saída. Com este avanço, o controlador, os sensores e os atuadores passaram a funcionar em conjunto, transformando processo em um sistema automatizado, onde o próprio controlador toma decisões em função da situação dos sensores e aciona os atuadores.

Os primeiros sistemas de automação operavam por meio de sistemas eletromecânicos, com relés e contatores. Neste caso, os sinais acoplados à máquina ou equipamento a ser automatizado acionam circuitos lógicos a relés que disparam as cargas e atuadores.

As máquinas de tear são bons exemplos da transição de um sistema de automação rígida para automação flexível. As primeiras máquinas de tear eram acionadas manualmente. Depois passaram a ser acionadas por comandos automáticos, entretanto, estes comandos só produziam um modelo de tecido, de padronagem, de desenho ou estampa.

A introdução de um sistema automático flexível no mecanismo de uma máquina de tear, tornou possível produzir diversos padrões de tecido em um mesmo equipamento. Com o avanço da eletrônica, as unidades de memória ganharam maior capacidade e com isso armazenam todas as informações necessárias para controlar diversas etapas do processo. Os circuitos lógicos tornaram-se mais rápidos, compactos e capazes de receber mais informações de entrada, atuando sobre um número maior de dispositivos de saída. Chegamos assim, aos microcontroladores responsáveis por receber informações das entradas, associá-las às informações contidas na memória e a partir destas desenvolver um a lógica para acionar as saídas.

Toda esta evolução nos levou a sistemas compactos, com alta capacidade de controle, que permitem acionar diversas saídas em função de vários sinais de entradas combinados logicamente.

Um outra etapa importante desta evolução é que toda a lógica de acionamento pode ser desenvolvida através de software, que determina ao controlador a seqüência de acionamento a ser desenvolvida. Este tipo de alteração da lógica de controle caracteriza um sistema flexível. Os CLPs são equipamentos eletrônicos de controle que atuam a partir desta filosofia.

3. Histórico

O Controlador Lógico Programável – CLP – nasceu dentro da General Motors, em 1968, devido a grande dificuldade de mudar a lógica de controle dos painéis de comando a cada mudança na linha de montagem. Tais mudanças implicavam em altos gastos de tempo e dinheiro.

Sob a liderança do engenheiro Richard Morley, foi preparada uma especificação que refletia as necessidades de muitos usuários de circuitos e relés, não só da indústria automobilística como de toda a indústria manufatureira.

Nascia assim um equipamento bastante versátil e de fácil utilização, que vem se aprimorando constantemente, diversificando cada vez mais os setores industriais e suas aplicações, o que justifica hoje um mercado mundial estimado em 4 bilhões de dólares anuais.

4. Vantagens

- menor espaço
- menor consumo de energia elétrica
- reutilizáveis
- programáveis
- maior confiabilidade
- maior flexibilidade
- maior rapidez na elaboração dos projetos
- interfaces de comunicação com outros CLPs e computadores

PRINCÍPIO DE FUNCIONAMENTO

Podemos apresentar a estrutura de um CLP dividida em três partes: entrada, processamento e saída.



Figura 1 – Estrutura básica de um CLP

Os sinais de entrada e saída dos CLPs podem ser digitais ou analógicos. Existem diversos tipos de módulos de entrada e saída que se adequam as necessidades do sistema a ser controlado.

Os módulos de entrada e saídas são compostos de grupos de bits, associados em conjunto de 8 bits (1 byte) ou conjunto de 16 bits, de acordo com o tipo da CPU.

As entradas analógicas são módulos conversores A/D, que convertem um sinal de entrada em um valor digital, normalmente de 12 bits (4096 combinações). As saídas analógicas são módulos conversores D/A, ou seja, um valor binário é transformado em um sinal analógico.

Os sinais dos sensores são aplicados às entradas do controlador e a cada ciclo (varredura) todos esses sinais são lidos e transferidos para a unidade de memória interna denominada memória imagem de entrada. Estes sinais são associados entre si e aos sinais internos. Ao término do ciclo de varredura, os resultados são transferidos à memória imagem de saída e então aplicados aos terminais de saída. Este ciclo esta representado na figura 2.

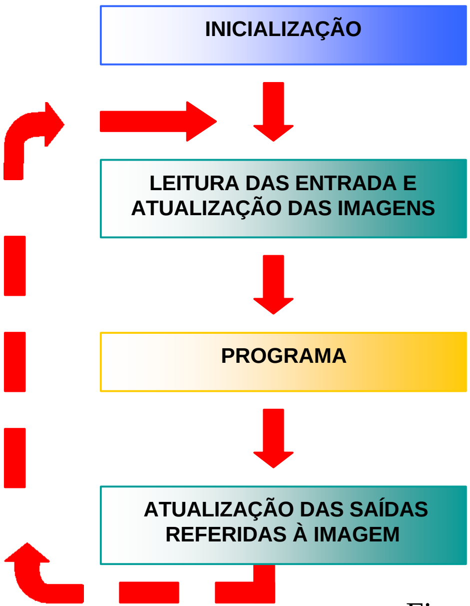


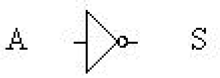
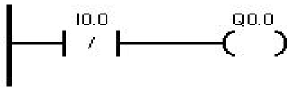
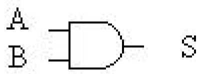
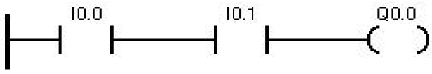
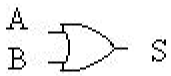
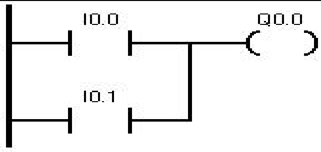
Figura 2 – Ciclo de processamento dos CLPs

INTRODUÇÃO A PROGRAMAÇÃO

1. Lógica matemática e binária

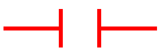
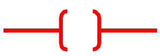
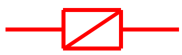
A lógica matemática ou simbólica visa superar as dificuldades e ambigüidades de qualquer língua, devido a sua natureza vaga e equívoca das palavras usadas e do estilo metafórico e, portanto, confuso que poderia atrapalhar o rigor lógico do raciocínio. Para evitar essas dificuldades, criou-se uma linguagem lógica artificial.

A lógica binária possui apenas dois valores que são representados por : 0 e 1. A partir desses dois símbolos construímos então uma base numérica binária. A partir desses conceitos foram criadas as portas lógicas, que são circuitos utilizados para combinar níveis lógicos digitais de formas específicas. Neste curso aprenderemos apenas as portas lógicas básicas: AND, OR e NOT.

Portas Lógicas	Símbolo	Expressão	Ladder
NOT		$S = \overline{A}$	
AND		$S = A \cdot B$	
OR		$S = A + B$	

Os CLPs vieram a substituir elementos e componentes eletro-eletrônicos de acionamento e a linguagem utilizada na sua programação é similar à linguagem de diagramas lógicos de acionamento desenvolvidos por eletrotécnicos e profissionais da área de controle, esta linguagem é denominada linguagem de contatos ou simplesmente LADDER.

A linguagem Ladder permite que se desenvolvam lógicas combinacionais, seqüenciais e circuitos que envolvam ambas, utilizando como operadores para estas lógicas: entradas, saídas, estados auxiliares e registros numéricos. A Tabela 1 nos mostra os 3 principais símbolos de programação.

Tipo	Símbolo	Equipamento elétrico
Contato aberto		
Contato fechado		
Saída		

Para entendermos a estrutura da linguagem vamos adotar um exemplo bem simples: o acionamento de uma lâmpada L a partir de um botão liga/desliga

Na figura 3 temos o esquema elétrico tradicional, o programa e as ligações no CLP.

Para entendermos o circuito com o CLP, vamos observar o programa desenvolvido para acender a lâmpada L quando acionamos o botão B1.

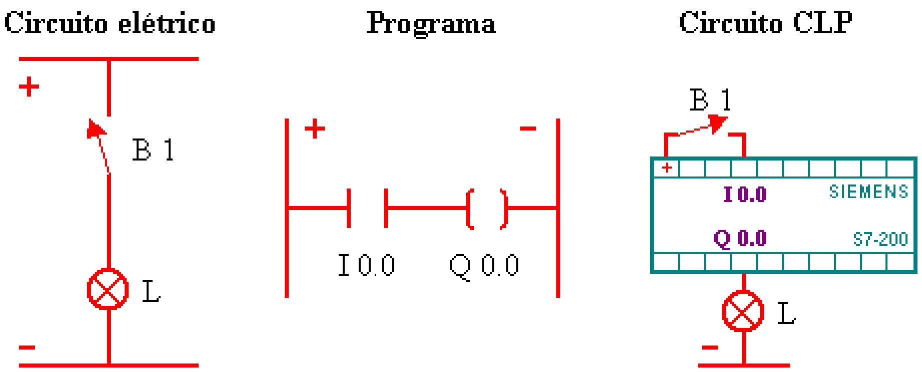


Figura 3 – Acionamento de uma lâmpada

O botão B1, normalmente aberto, está ligado a entrada I0.0 e a lâmpada está ligada à saída Q0.0. Ao acionarmos B1, I0.0 é acionado e a saída Q0.0 é energizada. Caso quiséssemos que a lâmpada apagasse quando acionássemos B1 bastaria trocar o contato normal aberto por um contato normal fechado, o que representa a função NOT.

Podemos desenvolver programas para CLPs que correspondam a operações lógicas combinacionais básicas da álgebra de Boole, como a operação AND. Na área elétrica a operação AND corresponde a associação em série de contatos, como indicado na figura 4.

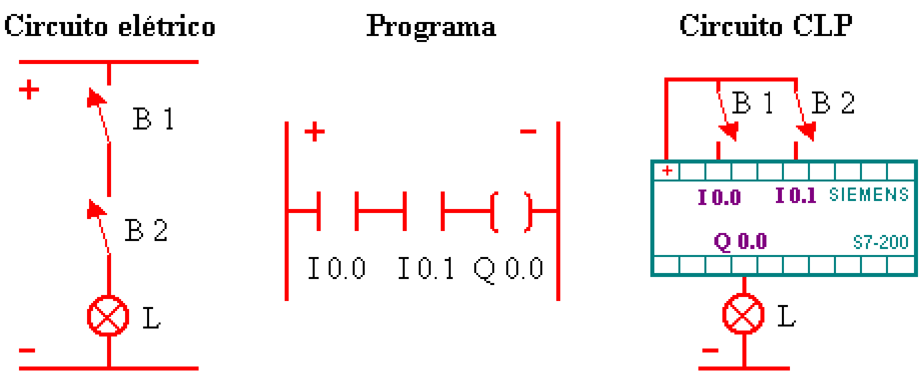


Figura 4 – Função AND

Outra operação lógica básica é a função OR, que corresponde a associação em paralelo de contatos, como indicado na figura 5.

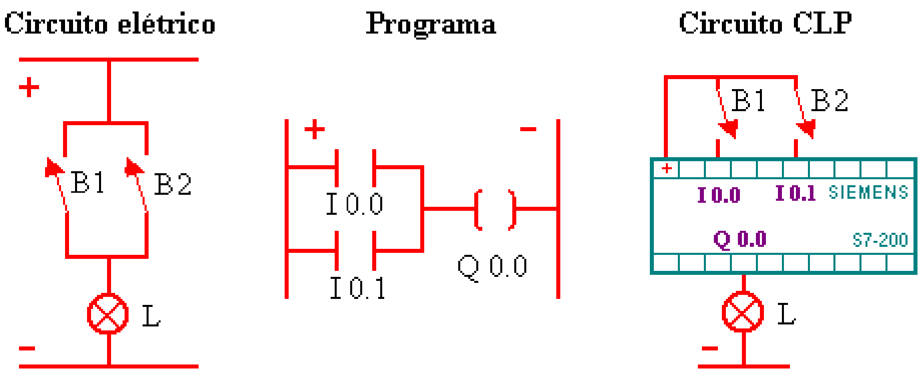


Figura 5 – Função OR

Assim podemos afirmar que todas as funções lógicas combinacionais podem ser desenvolvidas em programação e executadas por CLPs, uma vez que todas derivam dos básicos: NOT, AND e OR.

A flexibilidade dos CLPs é percebida neste momento pois as alterações lógicas podem ocorrer com grande facilidade, sem que sejam necessárias alterações do hardware ou inclusão de componentes eletrônicos ou elétricos. Esta é a principal característica dos sistemas de automação flexíveis e o que faz dos CLPs ferramentas de grande aplicação nas estruturas de automação.

Além da linguagem de contatos, existem outras formas de programação características de cada fabricante.

Concluímos então que os projetos de automação e controle envolvendo CLPs reduzem o trabalho de desenvolvimento de hardware dos circuitos lógicos do acionamento, bem como os dispositivos e potência para acionamento de cargas e dos atuadores, uma vez que podemos escolher módulos de saída já prontos, adequados ao tipo de carga que desejamos acionar.

A utilização desses controladores contemplam, por conseguinte alguns passos genéricos:

- *definição da função lógica a ser programada*
- *transformação desta função em programa assimilável pelo CLP*
- *implementação física do controlador e de suas interfaces com o processo*

Neste curso introdutório estaremos tratando da programação básica do CLP S7-200 através do programa STEP 7 para Windows. Serão consideradas apenas os recursos básicos, que são: contato normal aberto, contato normal fechado, contadores e temporizadores.

ACESSÓRIOS E NOVAS TECNOLOGIAS

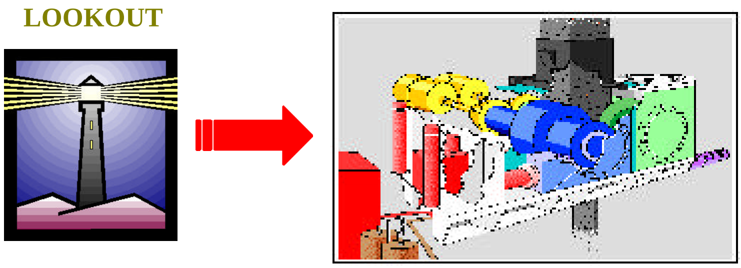
Os módulos de saída podem ser encontrados com drivers a transistor para carga DC, a relé para cargas AC e DC e a tiristores para cargas AC de potência.

As configurações de CLPs variam de fabricante a fabricante, e os módulos de entradas e saídas, sejam elas digitais ou analógicas, podem ser encontrados em grupos separados ou associados.

Existem também cartões de comunicação entre CLPs ou entre computadores, sejam eles industriais ou PCs. Estes cartões são muito utilizados e de extrema importância na automação de processos e máquinas, pois permitem que um sinal recebido por um PLC, possa acionar um contato de outro PLC ou de uma placa conversora A/D instalada em um computador, que estejam distantes.

Outro acessório importante é a IHM - Interface Homem-Máquina, que é uma painel de controle programável, que apresenta para o usuário mensagens de acordo com as condições dos sinais de entrada e saída, permitindo que um operador normal tome ciência da condição do sistema ou equipamento que está sendo controlado. Este acessório é utilizado como sistema supervisório e apresenta mensagens de emergência ou de parada por problemas técnicos.

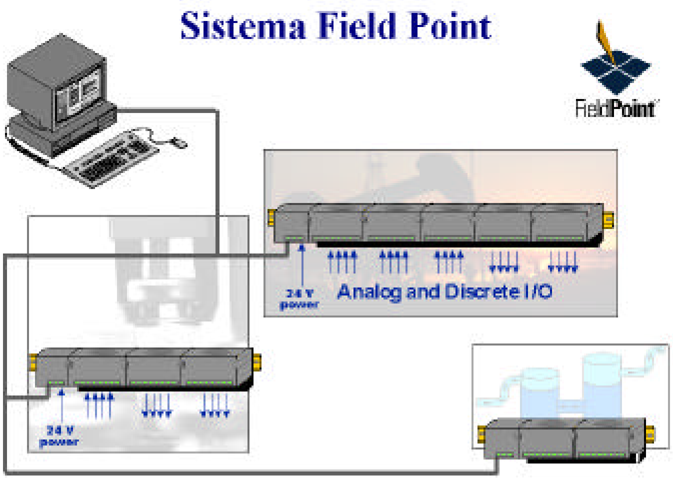
Atualmente estes painéis estão sendo substituídos por telas de computador, onde é possível reproduzir com grande perfeição o processo industrial, o que torna a interface com o operador muito mais amigável e segura. Dentre os softwares mais conhecidos no mercado estão o FIX e LookOut.



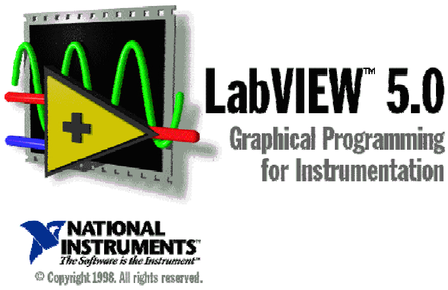
Conforme a tecnologia avança novos equipamentos e estruturas vão sendo inventados. Uma nova estrutura está sendo difundida e implantada que é a Field Bus, ou barramento de campo. Nesta nova estrutura os sensores e atuadores são interligados por um par trançado de fios o que torna a instalação mais barata. Para adicionar novos sensores e atuadores basta plugá-los neste barramento e reconfigurar o sistema, sem ter que gastar tempo e dinheiro para passar novos fios até a sala de controle, o que é muito complicado em instalações industriais. Essa estrutura é similar a de um computador onde existem vários slots e podemos conectar ou substituir diversos tipos de placas como: placas de vídeo, fax, controladoras de drivers, conversores A/D ou D/A, etc... Essa estrutura está sendo regulamentada e padronizada por associações internacionais como a Fieldbus Foundation

Fieldbus™ Foudation

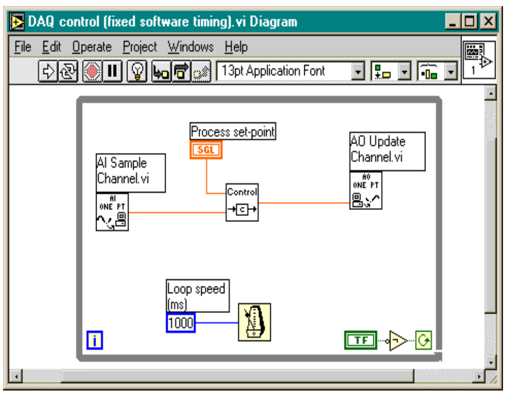
Um sistema similar é o Field Point. Esse sistema possui um computador central e o barramento com os módulos espalhados pelo chão de fábrica. Esses módulos possuem entradas e saídas analógicas e digitais. Possui vantagens similares ao Fiedbus, como a fácil instalação de módulos auxiliares e fácil modificação da estrutura de controle. Tudo é controlado via software instalado no computador central.



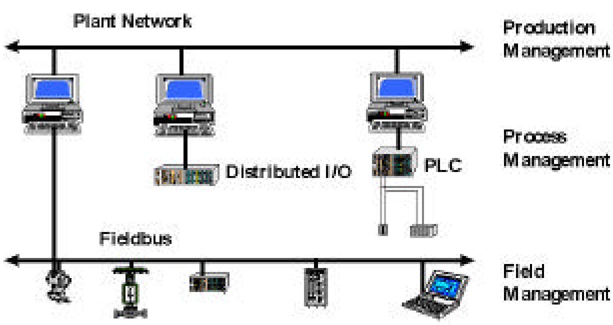
Uma outra área avança com grande força que é a instrumentação virtual, onde os sensores e atuadores são conectados a um computador e um software processa as informações, enviando posteriormente as respostas. As grandes vantagens desses softwares são: linguagem de programação gráfica, ferramenta de simulação interativa, aquisição de dados e controle, monitoramento e processamento de imagens. Dentre os softwares mais utilizados estão o LabView e o BridgeView.



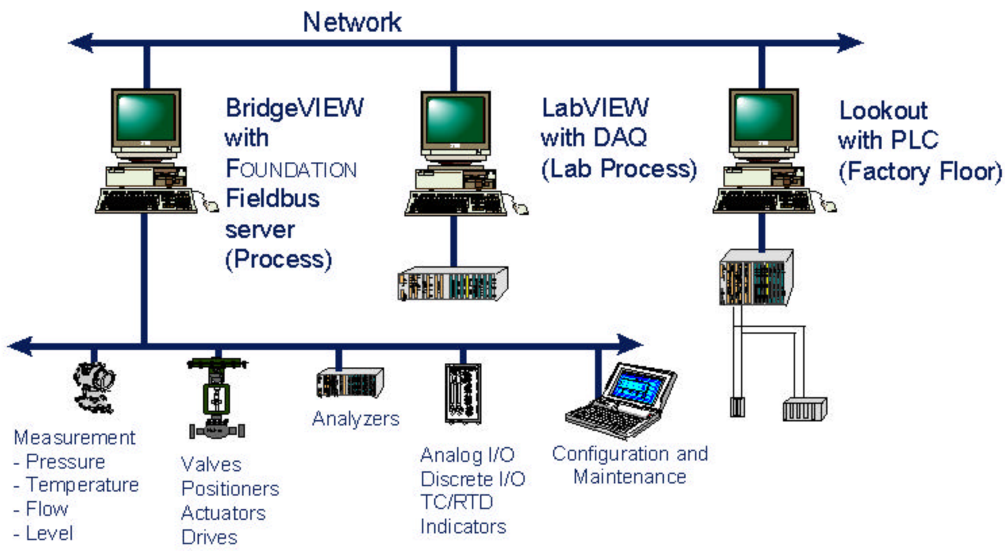
Programação Gráfica

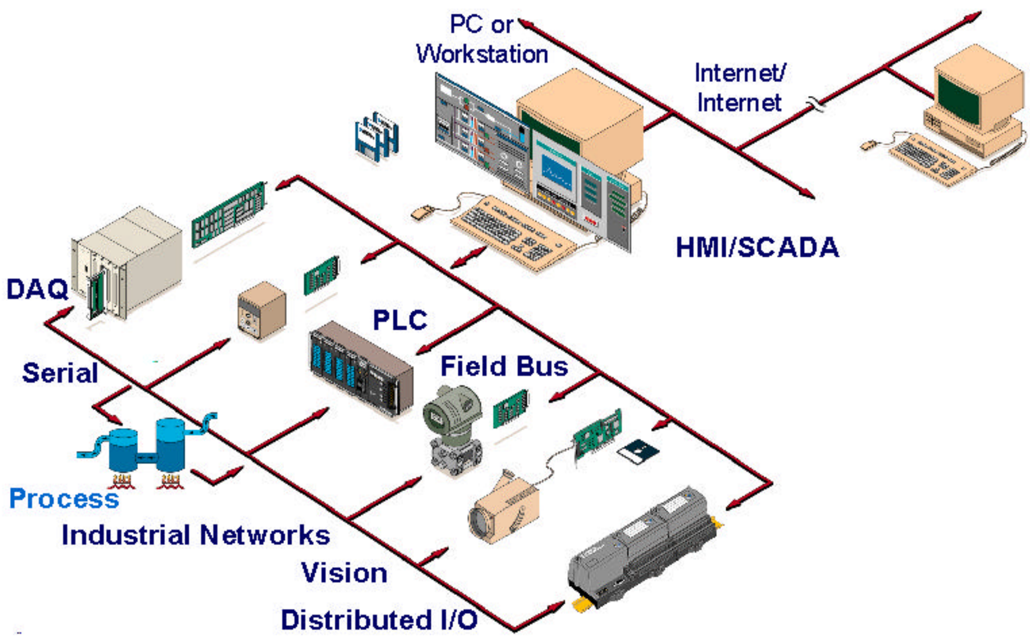


Arquitetura BridgeVIEW



Aplicações de Automação





SIEMENS SIMATIC S7-200

A família de controladores programáveis S7-200 foi desenvolvida para o controle de uma ampla gama de aplicações de controle e automação . Há varias opções de programação e a possibilidade de escolha dos equipamentos e da linguagem de programação .

O que é necessário para instalar o software ?

O software da SIEMENS S7-200 for Windows pode ser instalado em qualquer PC IBM ou compatível com , no mínimo , um processador Intel 386/33MHz ,08 MB RAM , display VGA color, HD com espaço livre de 35 MB e Windows 3.1, 3.11 ou 95.

Como é a comunicação do software ?

O Software da SIEMENS S7-200 comunica-se com a CPU S7-200 através da porta do programador na parte inferior da CPU . Você pode usar um cabo PC/PPI para conectar o seu programador nas comunicações online. O computador necessitará de uma porta serial RS-232-C de 09 pinos para fazer a comunicação. Caso o micro tenha uma porta serial DB-25 será necessário um adaptador para DB-09 .

Características do S7-200 - CPU 214

Dimensões : 197 x 80 x 62 mm

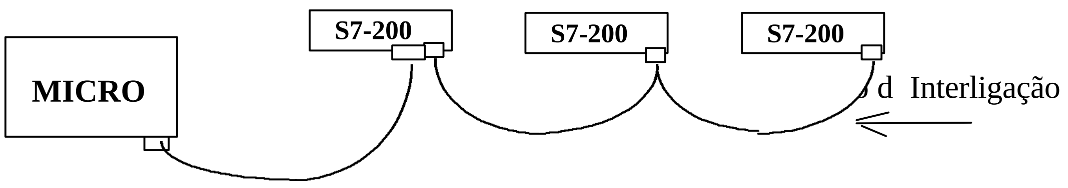
Fonte : 24 Volts (Faixa de Tensão: 20,4 à 28,8 Vcc)

Corrente Típica de Alimentação : 60 mA (Max. 500mA)

Entradas : 14 Portas	Estado ON (Faixa) : de 15 à 35 Vcc Estado OFF (Faixa) : de 0 à 5,0 Vcc
Tempo de Resposta :	I 0.0 à I 0.3 : 0,2 ms. I 0.4 à I 1.5 : 1,2 ms. I 0.6 à I 1.5 (usando HSC1 e HSC2): 30 µs
Saídas : 10 Portas	Corrente Máxima por Saída : 400 mA

Memória : 2 K Words / RAM autonomia 190 Hs
2 K Words / EEPROM (memória extra)

NOTA : Existe um cabo de comunicação especial da SIMENS que é utilizado para conectar mais de um CLP em um computador.



O que são Entradas e Saídas ?

I x.x - Designa uma entrada. É um elemento usado para monitorar uma ação ou um evento, como um interruptor, pressostato, termostato, etc.

Na CPU 214 nós temos 14 entradas digitais reais.

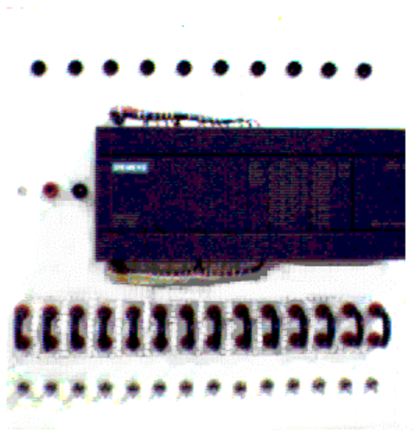
São elas: I 0.0, I 0.1, I 0.2, I 0.3, I 0.4, I 0.5, I 0.6, I 0.7, I 1.0, I 1.1, I 1.2, I 1.3, I 1.4, I 1.5.

Q x.x - Designa uma Saída. É usada para controlar um equipamento como um motor, uma válvula ou um LED.

Na CPU 214 nós temos 10 saídas digitais reais.

São elas: Q 0.0, Q 0.1, Q 0.2, Q 0.3, Q 0.4, Q 0.5, Q 0.6, Q 0.7, Q 1.0, Q 1.1.

10 SAIDAS DIGITAIS



14 ENTRADAS DIGITAIS

O que são Entradas e Saídas Lógicas e Analógicas ?

Entradas e saídas lógicas são aquelas que possuem apenas dois resultados, 0 e 1. Sendo o resultado 0 = 0V e o resultado 1 = 24V.

As Entradas e Saídas Analógicas podem variar passo a passo dentro de seu gradiente de variação.

Por exemplo: digamos que o Laboratório tenha um modulo adicional de 02 entradas +02 saídas analógicas 220Vca; Então estas entradas e saídas poderão variar suas tensões entre 0Vca e 220Vca assumindo valores tais como: 40V, 87V 152Vca.

O que são contatos de memória ?

Contatos de memória são entidades virtuais que são utilizados apenas para ajudar o desenvolvimento da lógica de programação escalar interna. Usam uma simbologia de entrada e de saída . ─┤ ├─ ─( )─

No caso da CPU mod. 214 , eles são 56 endereços variando do endereço M 0.0 ao endereço M 7.7 .

O que são entradas e saídas imaginárias?

Entradas e Saídas Imaginárias são aquelas que só podem ser usadas dentro do programa. Mas então elas deixam de ser Entradas e Saídas?
Sim, elas serão utilizadas para contatos internos do programa , a não ser que se instale um módulo adicional e então estas entradas e/ou saídas (depende do modulo) deixarão de ser imaginárias e se transformarão em reais.

No caso da CPU mod. 214, são elas :

Entradas: I 1.6 à I 7.7

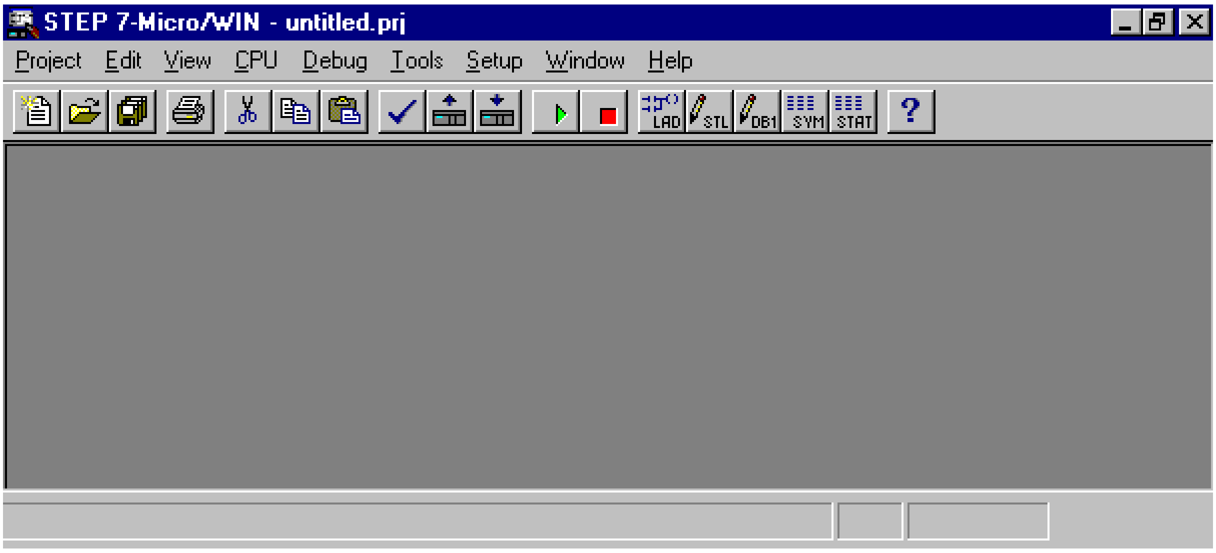
Saídas: Q 1.2 à Q 7.7

Nota: Os números que vem depois desses designadores identificam a entrada ou a saída específica que está sendo conectada ou controlada. Esses números vão de 0 a 7. Um grupo de oito pontos é chamado um BYTE. As Entradas e Saídas (I e Q), tem sua área de memória específica, assim uma entrada e uma saída podem ter o mesmo número de endereço, I 0.0 e Q 0.0.

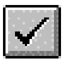
Por exemplo, se você deseja conectar um interruptor "liga/desliga" à terceira entrada é preciso liga-lo à entrada I 0.2.

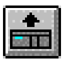
Tela de Abertura

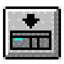
Depois de clicar no ícone, aparecerá em seu micro a tela de abertura . A partir deste ponto, você pode efetuar novos projetos, abrir projetos, alterar as configurações, etc.

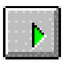


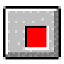
Como em outros programas para Windows, os menus são alterados dependendo da tarefa que você esteja executando. Esta tela possui uma barra de comandos e ferramentas com os comandos característicos do Windows como: novo arquivo, abrir arquivo, salvar, imprimir, recortar, copiar e colar. Nesta barra também temos outros ícones específicos que são os seguintes:

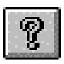
 Compile - Compilar o programa

 Upload - Ler o programa do CLP

 Download - Carregar o programa no CLP

 Run - Executar o programa

 Stop - Parar a execução do programa

 Help - Ajuda

Uma Visão Geral dos Menus

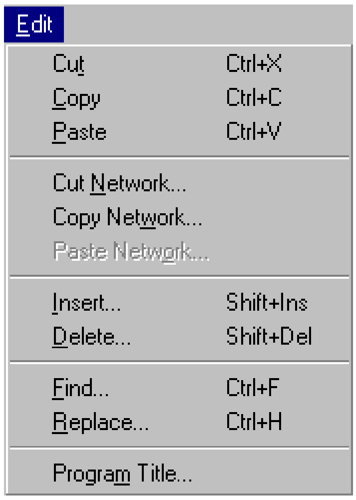
Na barra de comandos temos os menus: Project, Edit, View, CPU, Debug, Tools, Setup, Window e Help.

Os Menus Project e Edit

Em Project e Edit estão comandos similares ao do Windows e que estamos habituados a utilizar. Os comandos básicos são: new, open, close, save all, save as, import, export, page setup, print preview, print, print setup, exit, cut, copy , paste, find, replace, insert e delete.

Nestes menus também existem alguns comandos específicos como o download e o upload, que são para carregar e baixar programas do CLP; o cut network e o copy network, que são respectivamente para cortar e copiar uma linha de programação; e o program title que é para inserir o título do programa.

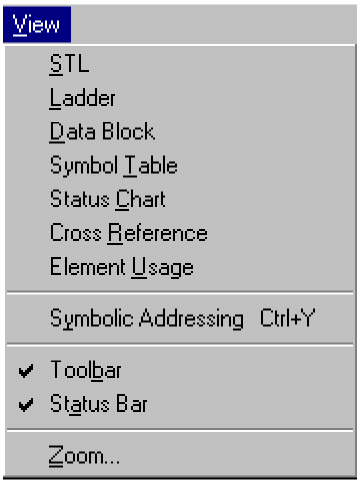




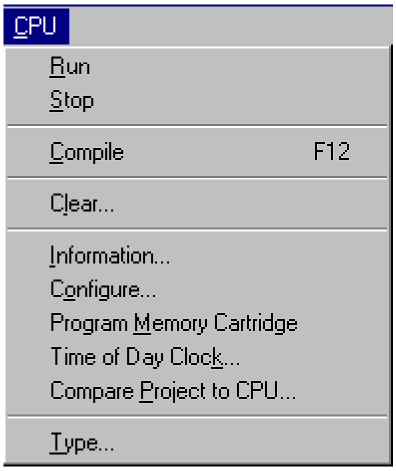
O Menu View

Neste menu estão as configurações de visualização das telas de programação.

Em View estão os comandos de seleção das barras de ferramenta: Toolbar e Status Bar. Nesta tela escolhemos o tipo de programação que se deseja utilizar. Pode-se escolher entre a linguagem Ladder e a STL, que é uma linguagem escrita. <u>Esta apostila adotará como linguagem padrão a Ladder, que é utilizada por todos os fabricantes de CLPs. Além disso, é uma linguagem gráfica, que é mais amigável e mais fácil.</u>

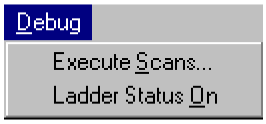


O Menu CPU



Neste menu estão os mesmos comandos do Toolbar que foram abordados anteriormente e mais alguns, que são: Clear, Information, Configure e Program Memory Cartridge. O Clear é para se apagar a memória e os outros comandos são para configurar e visualizar alguns parâmetros do CLP.

O Menu Debug

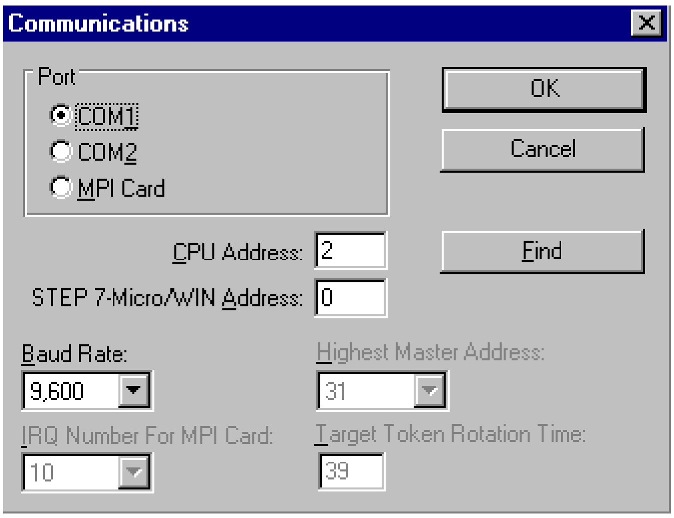


Em Debug, estão alguns comandos muito importantes e muito utilizados. O Execute Scans faz uma varredura em busca de um programa. O Ladder Status nos permite supervisionar o estado das entradas e saídas do CLP pela tela do micro. Com este comando ativado o operador pode monitorar todo o andamento do processo.

O Menu Setup

Em Setup estão as configurações de comunicação e de programação.



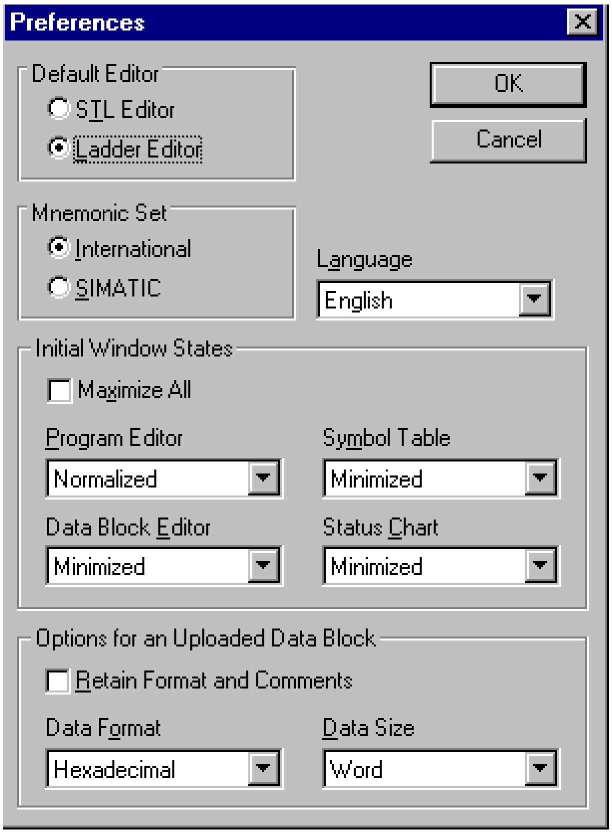


Em Communications configuramos a porta onde está conectado o cabo de comunicação do CLP e o endereço da CPU.

- CPU Address → Seleciona a estação que executará a função de controladora.

Exemplo:
001 → Esta estação é o próprio micro.
002 → Esta estação é o primeiro CLP.
003 → Esta estação é o segundo CLP.

Obs.: O LEE não possui o cabo que interliga mais de uma CLP. Então será sempre utilizado a estação 002.



Em preferences estão as configurações de programação. Podemos selecionar entre a programação STL e Ladder, padrão internacional ou Simatic

Podemos alterar o idioma do software e selecionar o estado inicial das telas de programação quando se inicia o software.

Alteramos também o formato e o tamanho para transferência de dados.

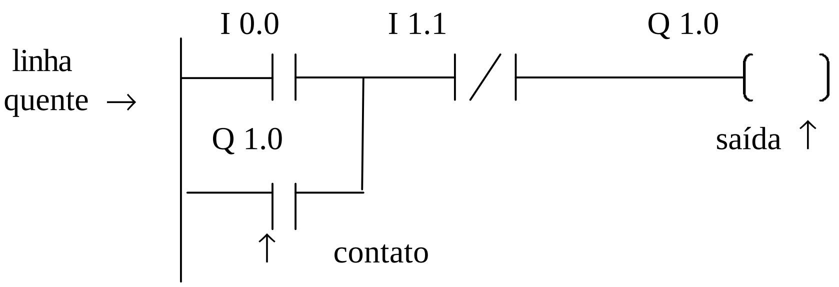
O Menu Help

Este menu de Ajuda oferece 3 caminhos diferentes para se obter o auxílio. No primeiro ele nos mostra todo o conteúdo. No segundo selecionamos a instrução que necessitamos de ajuda. O terceiro é dirigido aos usuários que utilizavam ou utilizam a versão DOS.



O que é a REDE de lógica escalar?

A figura abaixo mostra que a REDE da lógica escalar é uma fileira de elementos conectados que formam um circuito completo entre o trilho de força à esquerda e o elemento de saída à direita.



Nota: Observar que a energia flui da esquerda para direita.

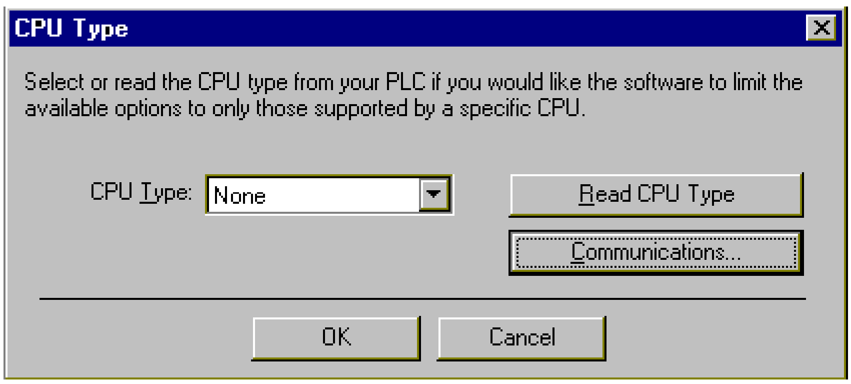
Vamos agora fazer o exemplo acima passo a passo.

Primeiramente vamos clicar em New Project,  na barra de ferramentas.

Aparecerá na tela uma janela onde devemos selecionar a CPU, que estamos utilizando, e as configurações de comunicação.

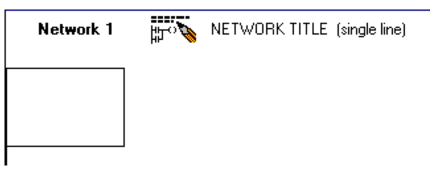
O comando Read CPU Type detecta automaticamente o tipo de CPU.

Em Comunications configuramos os parâmetros de comunicação, como foi abordado anteriormente.



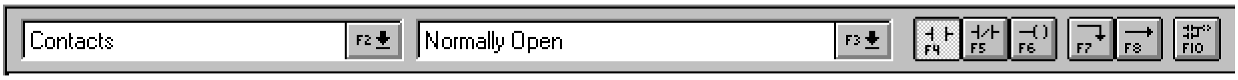
Após terminar as configurações clique em OK.

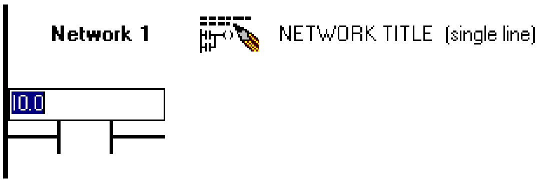
Nota: Observe que o cursor está parado na primeira coluna da primeira NETWORK; para movimentá-lo use as setas ←↑→↓, ou o mouse.



1º Passo: Inserir um contato normal aberto I 0.0.

Coloque o cursor na NETWORK 1 e selecione o contato normal aberto na barra de ferramentas. Para inserir o contato tecle ENTER ou dê um clique duplo. Acima do contato aparecerá um espaço para o endereçamento do contato.

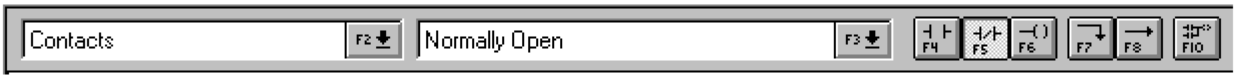




2º Passo: Digite o endereço do contato. Neste caso, como este contato tem o endereço 0.0, apenas tecle ENTER.

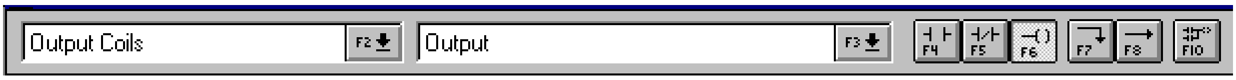
3º Passo: Vamos inserir agora o contato normal fechado I 1.1.

Coloque o cursor à direita e selecione o contato normal fechado na barra de ferramentas. Para inserir o contato tecle ENTER ou dê um clique duplo. Acima do contato aparecerá um espaço para o endereçamento do contato. Desta vez é necessário digitar o endereço I1.1 e teclar ENTER.

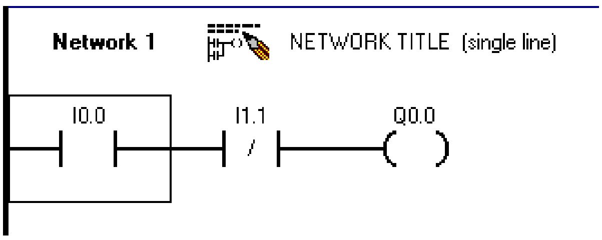


4º Passo: Vamos agora inserir uma saída Q 1.0.

Coloque o cursor à direita. Selecione a saída na barra de ferramentas e tecle ENTER ou dê um clique duplo. Acima da saída aparecerá um espaço para o endereçamento da mesma.



5º Passo – Vamos inserir agora o contato normal aberto Q 1.0, conhecido industrialmente como contato de selo. Para isto posicione o cursor no começo da network, selecione o comando Vert na barra de ferramentas e tecle ENTER ou dê um duplo clique. Cuidado porque para inserir esta linha vertical, o cursor deve estar posicionado corretamente, como no exemplo abaixo.



Para inserir o último contato, basta levar o cursor para baixo, selecionar o contato na barra de ferramentas e endereçar o mesmo como Q 1.0.

Blocos de Saídas Específicas (BOXs).

Vamos aprender agora outros tipos de blocos muito úteis para os programadores.
Existe uma quantidade grande de blocos e saídas específicas. Para visualizá-las, basta abrir as janelas F2 e F3 na barra de ferramentas.
Para saber a função de cada box indicado acima, basta consultar o Help.

Saída SET e RESET

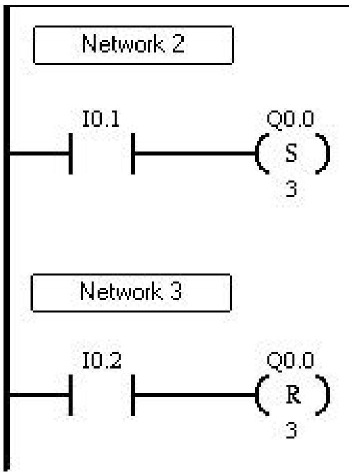
A saída SET nos permite acionar várias saídas simultaneamente.

Procedimento:

1º – Na Network2 insira um contato normal aberto I0.1

2º – Vamos levar o cursor até o fim da Network. Na janela F2 selecionar Output Coils e depois na janela F3 selecionar a saída SET.

Indique o número da saída.
Tecle Enter e embaixo da saída aparecerá um novo campo, onde selecionaremos as saídas a serem setadas.
Você pode selecionar o numero de saídas que você quiser.
No nosso exemplo, escolhemos K=3, a partir da saída Q 0.0. Então quando acionarmos a chave I 0.1, serão acionadas as saídas Q 0.0, Q 0.1 e Q 0.2.

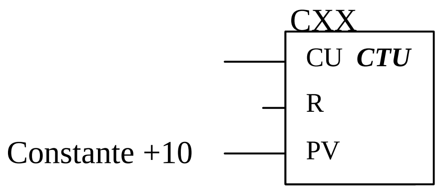


Nota: As saídas continuarão em ON mesmo que a chave 0.1 seja desligada.

Para voltar as saídas setadas pelo comando SET para OFF, será necessário fazer uma outra NETWORK usando o comando RESET.
Vamos agora construir a 3º NETWORK, repetindo os passos da anterior, sendo que no 1º passo, o endereço do contato aberto é I 0.2. Ao invés de SET, usaremos o RESET com K=3 a partir da saída Q 0.0.

C T U - Contador Crescente

Para selecioná-lo clique na Janela F2 em Timers e Counters e na Janela F3 em Count Up. Ele é composto de uma entrada "CU", um reset "R" e a constante a ser escolhida "PV". Esta constante define o número de vez que ele deverá contar para acionar a saída CXX.

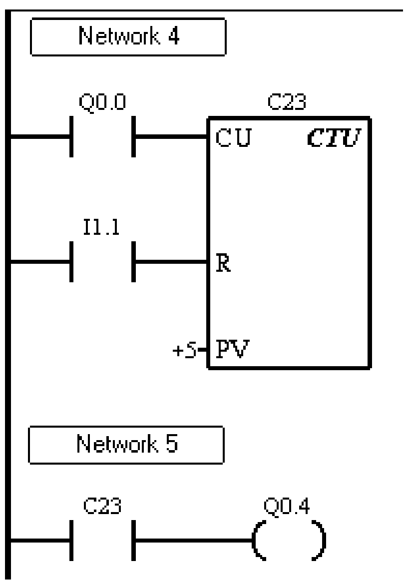


Nota: O CLP 214 pode endereçar vários contadores, isto é, podemos endereçar desde o contador C0 à C14 e do C80 ao C127.

A cada alteração de valor da entrada CU(0e1) o contador contará uma vez.

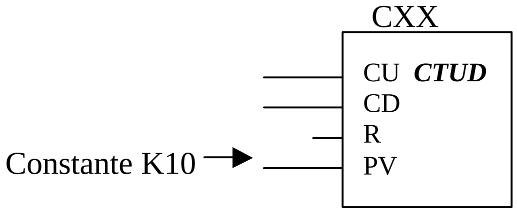
Por exemplo:

Vamos agora voltar ao nosso arquivo e editar mais uma NETWORK.
Desta vez você tentará fazer exatamente como mostra a figura anexa, sem ajuda.
Ao acionarmos cinco vezes a chave I 0.1, o contador será ativado e acionará o endereço C 23 que por sua vez na NETWORK 05 acionará a saída Q 0.4 .



CTUD - Contador Crescente e Decrescente

Para selecioná-lo clique na Janela F2 em Timers e Counters e na Janela F3 em Count Up/Down. Este contador é composto de uma entrada "CU", uma entrada "CD," um reset "R" e a constante a ser escolhida "PV".



Nota: O CLP214 pode endereçar vários contadores, isto é, podemos endereçar desde os contadores C48 ao C79.

OBS: A capacidade dos contadores vai de -32767 à 32767 eventos.

TON - Temporizador sem Paradas.

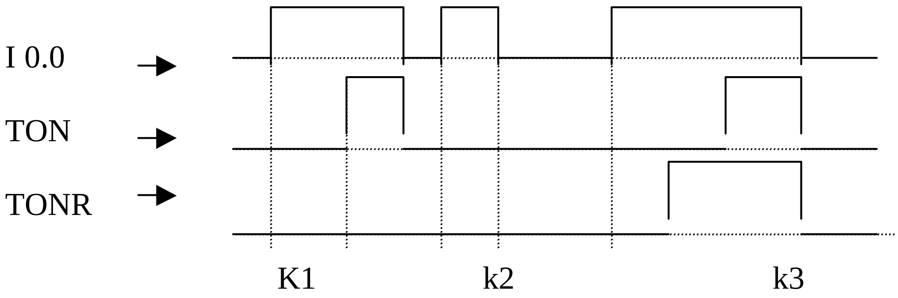
Para selecioná-lo clique na Janela F2 em Timers e Counters e na Janela F3 em Timer-On Delay.

Quando sua entrada "IN" é acionada, ele contará até que a mesma entrada seja desligada e se for re-ligada, ele voltará a contar do zero novamente.

TONR - Temporizador com Paradas.

Para selecioná-lo clique na Janela F2 em Timers e Counters e na Janela F3 em Timer-On Retentive Delay.

Quando sua entrada "IN" é acionada, ele contará até que a mesma seja desligada e se for re-ligada , o temporizador continuará a contar do mesmo ponto de onde parou.



T = K1+K2+K3=TONR

Estes temporizadores são compostos de uma entrada(IN) e uma constante ”PT”.
A constante “PT” deverá ser “K=xxxx”, sendo que deve ser respeitado a seguinte condição:

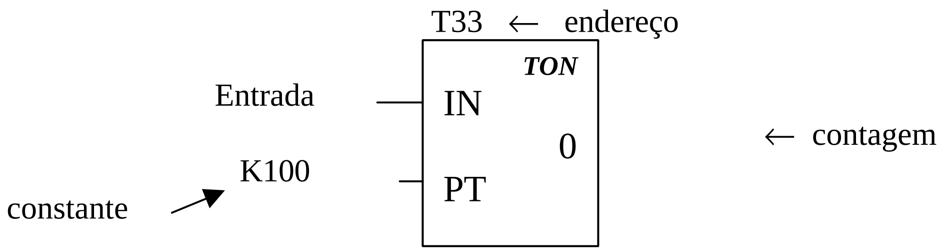
Tipo	Constante de tempo	Tempo ma	Endereços
TONR	01mS	32,76S	T0 e T64
TON			T32 e T96
TONR	10mS	327,67S	T1 à T4 e T65 à T68
TON			T33 à T36 e T97 àT100
TONR	100mS	3276,7S	T5 à T31 e T69 à T95
TON			T37 à T63 e T101 à T127

Exemplos:

TON T33 K=100 _ O TON tem constante de tempo igual a 10mS e se K=100, então teremos: 100×10mS=1S

TONR T29 K=600 Teremos: 600×100mS=60S

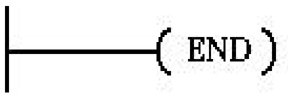
TON T64 K=20000 Teremos: 20000×1mS=20S



END

Este comando avisa ao CLP o final da programação, por isso de vital importância em qualquer programa. Este comando está na Janela F2 em Program Control.

Vamos tentar implementar aquele nosso exercício de demonstração sem o comando END no final da última linha. Irá aparecer uma mensagem de erro, lhe indicando que faltou o END.



GUIA DE PROGRAMAÇÃO

Para se programar em um CLP é importante seguir os seguintes passos:

1º – Rascunhar sua programação de contatos no papel para ajudar na implementação com o micro .

2º – Batizar o programa, para isto basta na tela de abertura salvá-lo com um nome apropriado.

3º – Digitar toda sua edição e salvá-la também em diskette, para garantir um backup.

4º – Gravar na memória do CLP o programa gravado no micro .

5º – Simulação em tempo real do programa no CLP com monitoração do micro .

6º – Efetuar as correções necessárias.

Como implementar o seu programa?

O primeiro passo deve ser o de apagar o programa residente na memória do CLP. Em seguida, deve-se compilar o seu programa e gravá-lo no CLP.

Como fazer isto?

Estas operações são muito simples e para isso basta no Menu CPU clicar em Clear... . Para compilar o programa basta clicar no ícone Compile e para gravá-lo no CLP basta clicar na barra de ferramentas, no ícone DownLoad.

- **<u>Clear CLP Memory</u>:**

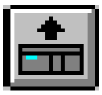
É utilizado para limpar da memória do CLP o programa residente. Este comando se encontra no menu CPU .

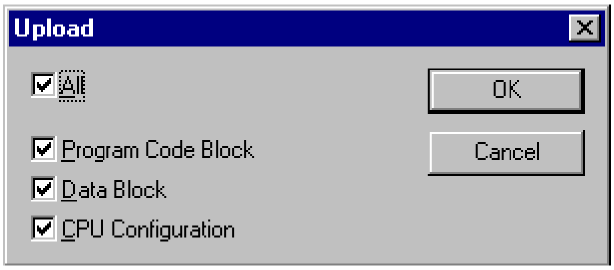
- **<u>Compile</u>:**



É utilizado para compilar o programa. Quando se faz a compilação o software faz uma varredura no programa em busca de erros, como por exemplo a falta do END no final do programa. Este comando se encontra na barra de ferramentas e no menu CPU .

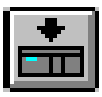
- **<u>UpLoad from CLP</u>**

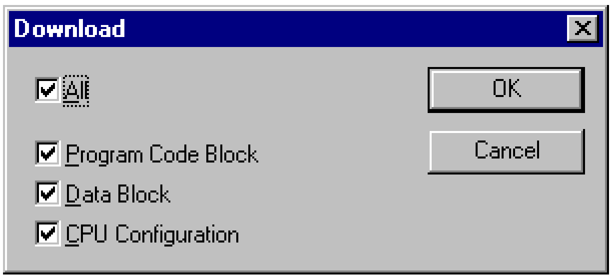




É utilizado para deslocar o que está na memória do CLP para o computador.

- **<u>DownLoad to CLP</u>**





Este comando é utilizado para baixar o programa que está no micro para a memória do CLP.

- ✓ Program Block - Transfere apenas o diagrama de contatos para o CLP.
- ✓ Data Block - Transfere apenas dados do programa tais como comentários e títulos.
- ✓ System Memory - Transfere apenas variáveis de memória.

<u>RUN e STOP</u>

Estes comandos citados acima só funcionam se o CLP estiver em modo de operação STOP.

O que vem a ser isto?

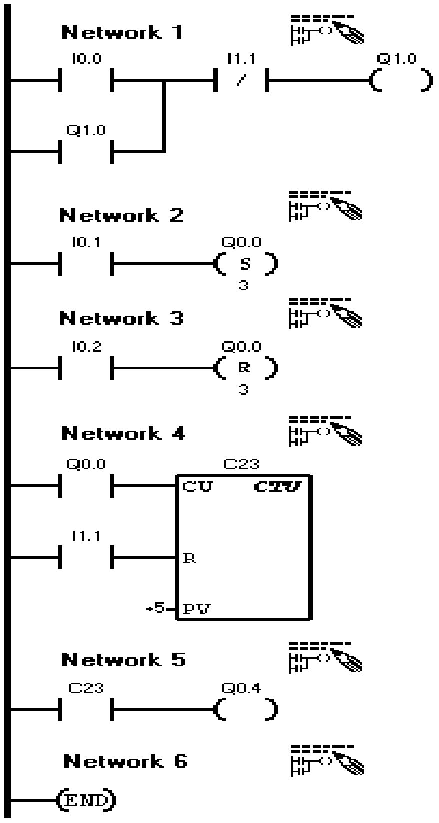
O CLP pode se encontrar em dois estados: parado (STOP), sem processar o programa residente ou em processamento (RUN), processando os dados de entradas e saídas.

Experimente implementar um programa já pronto do diretório C:\S7\Programs\

Estes programas mostrados na figura acima são programas que já vieram no software de instalação S7200 da Siemens. Escolha algum arquivo para exemplo e depois implemente-o no CLP.

Monitorando o Sistema (Ladder Status)

Agora vamos monitorar o programa implementado no CLP pelo micro.
Se o programa de demonstração foi implementado com sucesso ficou da seguinte forma:



Com este programa implementado você poderá monitorar as mudanças de estados das entradas e saídas pelo micro utilizando o comando Ladder Status On, que se encontra no menu principal Debug.

Vamos nos certificar que o CLP esta em modo **RUN** para podermos continuar.

Você observa que o contato normal fechado **I 1.1** está grifado mostrando continuidade naquele trecho .

Ao acionarmos a chave I 0.0 teremos continuidade no contato normal aberto I 0.0 assim energizando a saída Q 1.0 e fechando seu contato Q 1.0 , como mostra a figura abaixo .

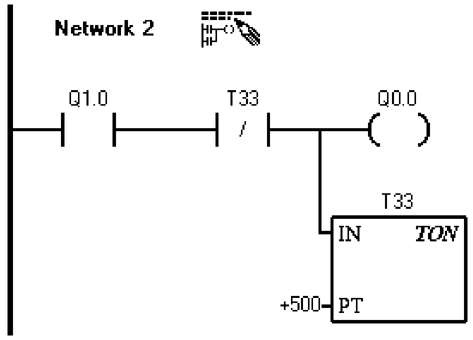
Com este comando STATUS , você agora já consegue monitorar qualquer programa sendo executado na memória do CLP .

Como modificar o programa?

Para fazer mudanças no seu programa , basta sair da função STATUS , pressionando no Menu Debug , Ladder Status OFF.

No programa de demonstração vamos fazer as seguintes alterações :

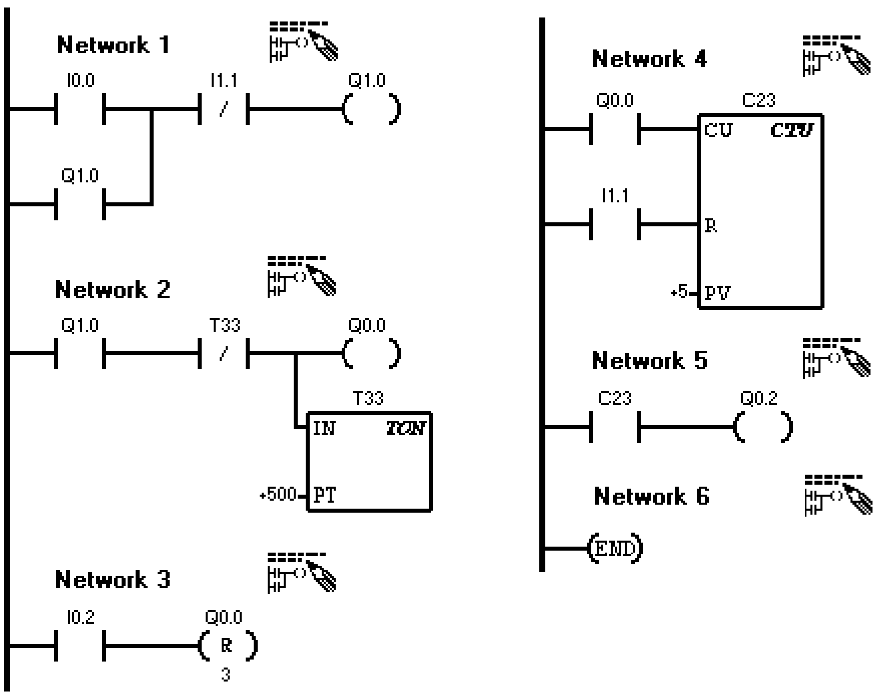
Na Network 2 vamos renomear o endereço do contato normal aberto I 0.1 para Q 1.0 , inserir o contato normal fechado de endereço T 33 e trocar a saída SET, por uma saída normal Q 0.0. Inserir em paralelo com a saída, um temporizador de endereço T 33, como nos mostra a figura:



Observando a figura vamos ver que o temporizador está preparado para disparar em 5 segundos, como já foi visto na página 23.

Na Network 5 vamos apenas renomear a saída Q 0.4 para Q 0.2 .

Então o programa estará da seguinte forma :



Você pode não ter percebido com essas mudanças, você montou um pequeno circuito temporizado.

Ao acionarmos I 0.0, acionaremos a saída Q 1.0, assim energizaremos o seu contato normal fechado na network 02 e energizando a saída Q 0.0 e o temporizador T33.

Após o tempo de 5 seg., o próprio contato T33 do temporizado o resetará, reiniciando do zero, isto é, fazendo o temporizador se transformar em um oscilador de T = 5 seg.

Na Network 04 o contador C 23 contará os pulsos gerados pelo contato normal aberto Q 0.0, quando o mesmo contar 5 pulsos acionará a saída Q 0.2 .

O que toda esta lógica de contatos significa?

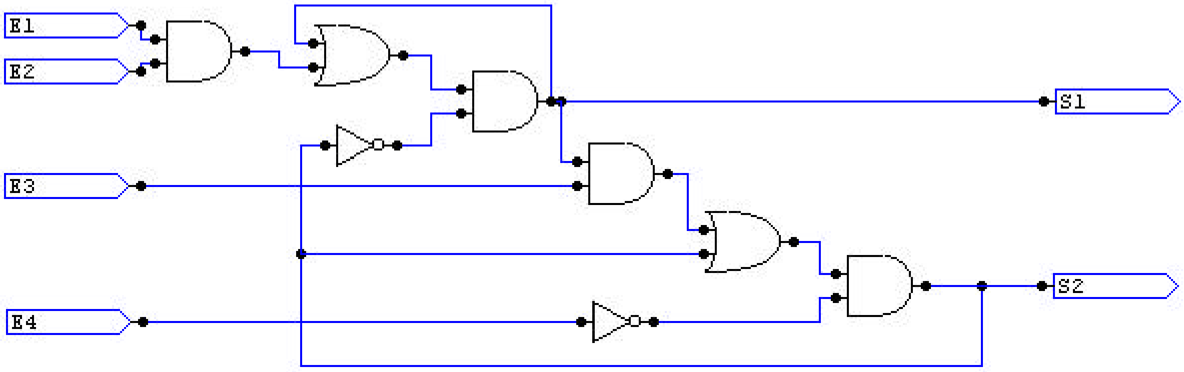
Imagine que você é um operador de uma refinaria, e este circuito será um circuito de alarme de uma caldeira. Sendo os endereços abaixo as seguintes descrições.

I 0.0 - Sensor de válvula de pressão
I 0.2 - Reset da saída
I 1.1 - Sensor de normalização da pressão
Q 0.0 - Válvula aliviadora de pressão de emergência
Q 0.2 - Desligamento de emergência
Q 1.0 - Alarme

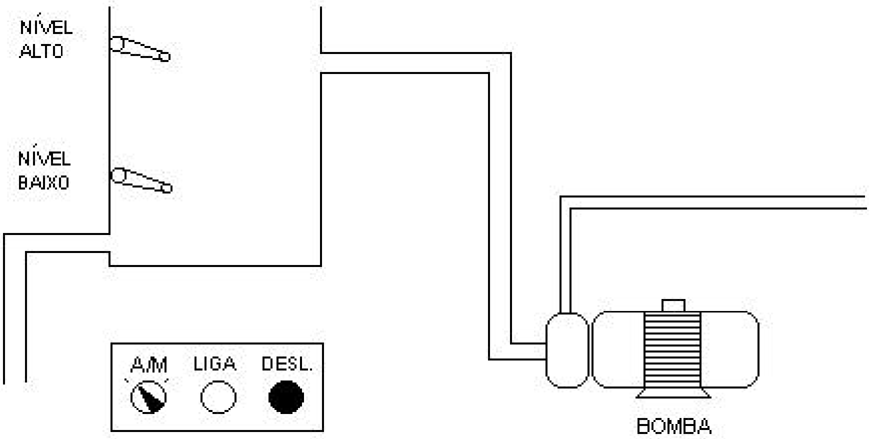
Quando a válvula de pressão for acionada o alarme é acionado imediatamente e a válvula aliviadora. Também se após 25 segundos o circuito não se normalizar o contador desliga todo o processo.

EXERCÍCIOS

1) Dada a lógica de comando digital abaixo, escreva um programa equivalente para CLP em linguagem Ladder. (Questão do Exame Nacional de Cursos 1998)



2) Desenvolver um projeto de controle para a seguinte instalação (na linguagem LADDER):



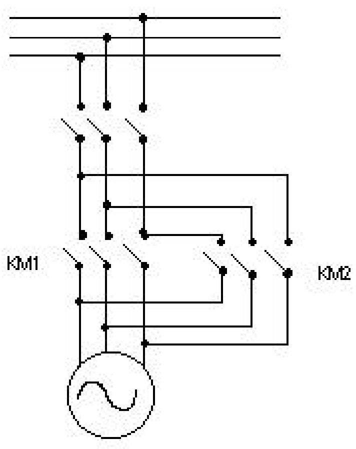
Através do programa o utilizador deve ser capaz de selecionar o modo se funcionamento : Automático ou Manual .
Em MANUAL , a Bomba poderá ser ligada pressionando-se o botão LIGA e desliga pressionando-se o botão desliga . Neste modo , as bóias de Nível não tem nenhuma ação .
Em AUTOMÁTICO, a bomba será ligada 10 Seg. após a deteção de NÍVEL BAIXO e desligada 10 Seg. após a deteção de NÍVEL ALTO .

ENTRADAS: I0.0 = 1 se NÍVEL < NÍVEL BAIXO - I0.0 = 0 se NÍVEL > NÍVEL BAIXO.
I0.1 = 1 se NÍVEL > NÍVEL ALTO - I0.1 = 0 se NÍVEL < NÍVEL ALTO.
I0.2 = 1 se AUTOMÁTICO - I0.2 = 0 se MANUAL .
I0.3 = 1 se BOTÃO LIGA pressionado .
I0.4 = 0 se BOTÃO DESLIGA pressionado .

SAÍDA: Q0.1 = 1 então BOMBA LIGADA .

3) Projete um controle capaz de inverter o sentido de rotação de um motor trifásico.

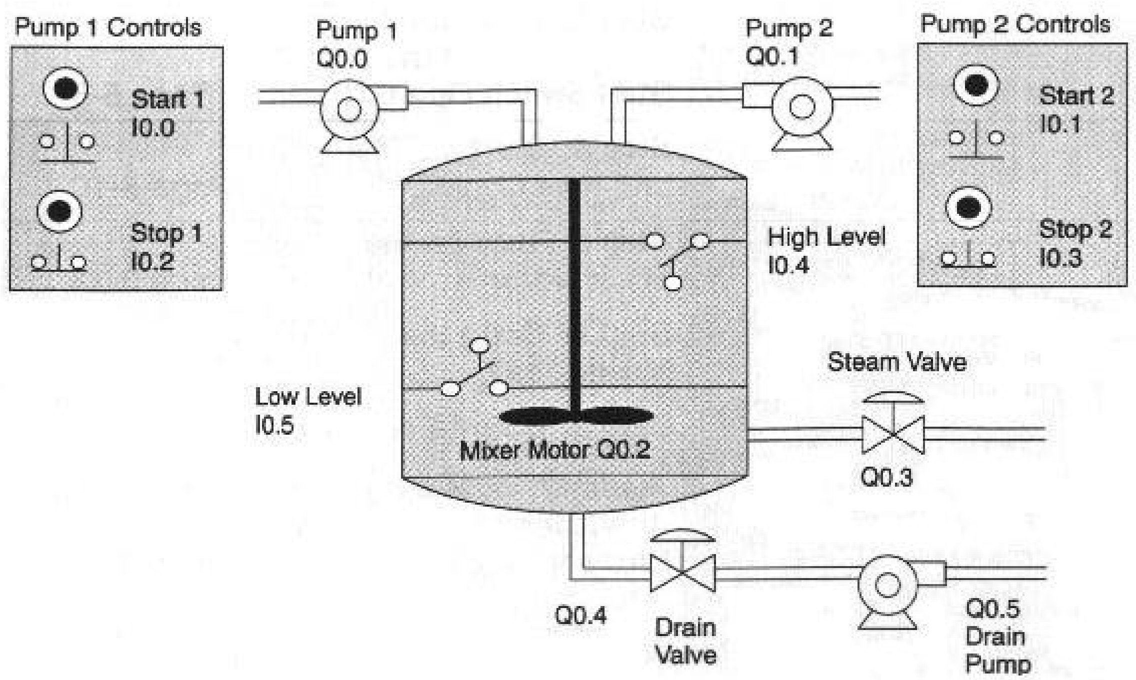
OBS: Para mudarmos o sentido de rotação de um motor trifásico é necessário que mudemos duas das três fases , isto é , que a fase A se torne B e que a fase B se torne A .



PS. Fazer : I0.0 = Botão para ligar
I0.1 = Botão de emergência
I0.2 = Acionamento frente
I0.3 = Acionamento ré
Q0.0 = Chave KM2
Q0.1 = Chave KM1

4) A figura abaixo mostra um misturador usado para fazer cores personalizadas de tinta.
Possuem dois encanamentos entrando no topo do tanque , fornecendo dois ingredientes diferentes , e um único encanamento no fundo do tanque para transportar a tinta misturada finalizada. Nessa aplicação você vai controlar a operação de preenchimento , monitorar o nível do tanque , e controlar o misturador e o período de aquecimento . Seguir os passos 1 até o 8 listados abaixo .

1º passo – Encha o tanque com o ingrediente 1.
2º passo – Encha o tanque com o ingrediente 2.
(a utilização do 1º ou do 2º ingrediente são independentes)
3º passo – Monitore o nível do tanque para o acionamento da chave "High-Level", utilizando um sensor de nível .
4º passo – Manter o status da bomba se a chave "Start" está aberta , isto é , a chave "start" deve ser independente (também perceba que o contato a ser utilizado deve ser normal fechado) .
5º passo – Comece a misturar os ingredientes e o período de aquecimento (10 Seg. por exemplo).
6º passo – Ligue o motor do misturador e a válvula de vapor (através destes haverá a mistura e aquecimento , respectivamente) .
7º passo – Drene o tanque da mistura através da válvula "Drain Valve"(válvula de drenagem) e do motor "Drain Pump"(bomba de drenagem).
8º passo – Crie um modo de contar quantas vezes este processo (descrito do 1º ao 7º passo) é realizado por completo .



DESAFIO

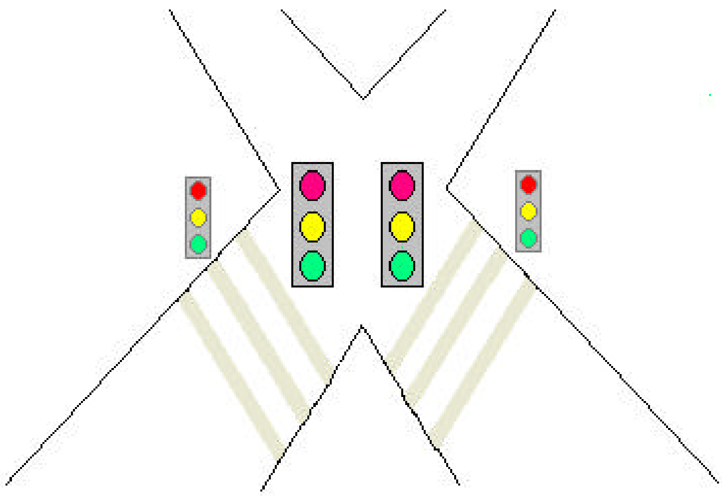
5) Projete e implemente no CLP em linguagem LADDER o controle de sinalização de um cruzamento de duas ruas. O cruzamento possui em cada rua, um sinal para pedestres e um para o automóveis.

PARÂMETROS

I0.0 – Liga
I0.1 – Desliga
Q0.0 – Verde (1º rua)
Q0.1 – Amarelo (1º rua)
Q0.2 – Vermelho (1º rua)
Q0.3 – Verde (2º rua)
Q0.4 – Amarelo (2º rua)
Q0.5 – Vermelho (2º rua)
Q0.6 – Verde (Pedestre 1º rua)
Q0.7 – Vermelho (Pedestre 1º rua)
Q1.0 – Verde (Pedestre 2º rua)
Q1.1 – Vermelho (Pedestre 2º rua)

SINAL	TEMPO (SEG.)	
	1º RUA	2º RUA
VERDE	10	10
AMARELO	5	5
VERMELHO	17	16

DICA : Se o sinal para carros estiver verde ou amarelo , o sinal de pedestres deve estar vermelho.



BIBLIOGRAFIA

- Bignell, J. W. e Donovan, R. L. – Eletrônica Digital – Editora Makron Books
- Bolton, W. – Engenharia de Controle – Editora Makron Books
- Castrucci, P. B. L. e Batista, L. – Controle Linear – Editora Edgar Blucher Ltda.
- Medeiros Júnior, Jair – Mafra, Marcos Augusto – Manual de utilização de Controladores Lógicos Programáveis – SIMATIC S7-200
- Ogata, Katsumi – Engenharia de Controle Moderno – Editora Prentice Hall do Brasil
- Osborne, A – Microprocessadores – Editora Mc Graw-Hill
- Revista Saber Eletrônica – nº 303 a 306 – Editora Saber
- Siemens – S7-200 Programmable Controller – Quick Start (Primer)
- Treinamento básico em CLP's Mitsubishi – Famiília FX